Amendments to the Specification:

Please replace paragraph 0004 with the following:

In recent years the OS user level binaries, however, have seen many virus and Trojan attacks. In these attacks a malicious user, software program, or the like, may modify an OS user level binary to gain illegal access to a computer, or inflict damage to the computer system itself. Currently, many administrators of these computer systems do not have the necessary mechanisms in place to detect an OS user level binary tampering by a malicious user. Thus, there is a need in the industry to provide a mechanism for detecting tampering of at least OS user level binaries. Therefore, it is with respect to these considerations, and others, that the present invention has been made.

Please replace paragraph 0029 with the following:

As shown in the figure, secure system 200 includes protected operating system 220-and user applications 108. Protected operating system 220 includes protected user level binaries 208-210, and kernel 202. Kernel 202 includes OS tamper detector 206 and tamper store 204. OS tamper detector 206 is in communication with tamper store 204 ~~202~~ and protected user level binaries 208-210.

Please replace paragraph 0031 with the following:

Tamper store 204 ~~202~~ is configured to provide storage and access to the selected integrity data for protected user level binaries 28-210. Tamper store 204 ~~202~~ may also include integrity data associated with kernel 202. Tamper store 204 ~~202~~ may be implemented employing a variety of mechanisms, including, but not limited to, a database, folder, file, program, and the like. In one embodiment tamper store 204 ~~202~~ is embedded within kernel 202 to minimize access by programs other than kernel 202. In another embodiment tamper store 204 ~~202~~ is encrypted using any of a variety of symmetric, and asymmetric key encryption algorithms. In yet another

embodiment, the integrity data is digitally signed prior to placing it into tamper store 204 ~~202~~, with an encryption key strongly associated with the kernel 202.

Please replace paragraph 0032 with the following:

While tamper store 204 ~~202~~ is illustrated as a component external to OS tamper detector 206, the present invention is not so limited. For example, tamper store 204 ~~202~~ may be included in OS tamper detector 206, located elsewhere, and the like, without departing from the scope or spirit of the present invention.

Please replace paragraph 0033 with the following:

OS tamper detector 206 is operable to examine data associated with OS user level binary 208-210 and determine whether it has been modified. OS tamper detector 206 may do so by performing actions substantially as described below in conjunction with FIG. 5. Briefly, however, OS tamper detector 206, may receive the data about the integrity of OS user level binary (208-210), and compare the received data against associated integrity data stored in tamper store 204 ~~202~~. In one embodiment, OS tamper detector 206 is configured to examine the integrity of a OS user level binary (208-210) during a read, write, and other specified operations are requested by the OS user level binary (208-210) upon an OS partition.

Please replace paragraph 0042 with the following:

Computer system 300 also includes input/output interface 324 for communicating with external devices, such as a mouse, keyboard, scanner, or other input devices not shown in FIG. 3. Likewise, computer system 300 may further include additional mass storage facilities such as CD-ROM/DVD-ROM drive 326 and hard disk drive 328. Hard disk drive [[2]]328 is utilized by computer system 300 to store, among other things, application programs, databases, and the like.

Please replace paragraph 0056 with the following:

It will be understood that each block of the flowchart illustration, and combinations of blocks in the flowchart illustration, can be implemented by computer program instructions. These program instructions may be provided to a processor to produce a machine, such that the instructions, which execute on the processor, create means for implementing the actions specified in the flowchart block or blocks. The computer program instructions may be executed by a processor to cause a series of operational steps to be performed by the processor to produce a computer implemented process such that the instructions, which execute on the processor, provide steps for implementing the actions specified in the flowchart block or blocks.